PREMIUM AUDIT

NAME OF PROJECT	DAOBAO
TOKEN BEP20 ADDRESS	0x21982f0a78e8248f7318aa7dd6c19daac9018ecb
WEBSITE / TELEGRAM	https://daobao.tech

DISCLAIMER: PLEASE READ FULL AUDIT

IMPORTANT DISCLAIMER

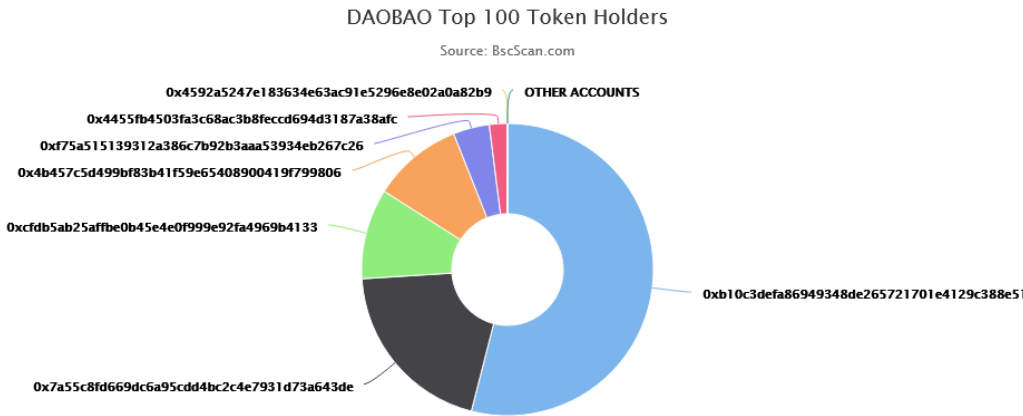|		
I.	**NOT RESPONSABLE**	Our audit report is not responsible for anything if the project audited turn into a scam.
II.	**NOT GUARANTEE**	Analytix Audit in no way guarantees that a project will not remove liquidity, sell off team tokens, or exit scam.
III.	**INFORMATION**	Analytix Audit do a research and provides public information about the project in an easy-to-understand format for the common person.
IV.	**AUDIT AGREE**	Agreeing to an audit in no way guarantees that a team will not remove all liquidity ("Rug Pull"), remove liquidity slowly, sell off tokens, quit the project, or completely exit scam
V.	**ENCOURAGE INVESTMENT**	Analytix Audit in no way takes responsibility for any losses, nor does Analytix Audit encourage any speculative investments
VI.	**INFORMATION**	The information provided in this audit is for information purposes only and should not be considered investment advice. Analytix Audit does not endorse, recommend, support, or suggest any projects that have been audited.

TOKENOMICS - OVERVIEW

The top 100 holders collectively own 100.00% (5,000,000,000.00 Tokens) of DAOBAO   Token Total Supply: 5,000,000,000.00 Token | Total Token Holders: 7

DAOBAO Top 100 Token Holders

Source: BscScan.com

0x4592a5247e183634e63ac91e5296e8e02a0a82b9   OTHER ACCOUNTS
0x4455fb4503fa3c68ac3b8feccd694d3187a38afc
0xf75a515139312a386c7b92b3aaa53934eb267c26
0x4b457c5d499bf83b41f59e65408900419f799806

0xcfdb5ab25affbe0b45e4e0f999e92fa4969b4133

0xb10c3defa86949348de265721701e4129c388e51

0x7a55c8fd669dc6a95cdd4bc2c4e7931d73a643de

(A total of 5,000,000,000.00 tokens held by the top 100 accounts from the total supply of 5,000,000,000.00 token)

Rank	Address	Quantity (Token)	Percentage
1	0xb10c3defa86949348de265721701e4129c388e51	2,699,999,998.56	54.0000%
2	0x7a55c8fd669dc6a95cdd4bc2c4e7931d73a643de	1,000,000,000	20.0000%
3	0xcfdb5ab25affbe0b45e4e0f999e92fa4969b4133	500,000,000	10.0000%
4	0x4b457c5d499bf83b41f59e65408900419f799806	500,000,000	10.0000%
5	0xf75a515139312a386c7b92b3aaa53934eb267c26	200,000,000	4.0000%
6	0x4455fb4503fa3c68ac3b8feccd694d3187a38afc	100,000,000	2.0000%
7	0x4592a5247e183634e63ac91e5296e8e02a0a82b9	1.44	0.0000%

(This comment refers to when the audit was carried out maybe team locked tokens later)

CONTRACT CODE - OVERVIEW

```
520      * Requirements:
521      *
522      * - the calling contract must have an ETH balance of at least `value`.
523      * - the called Solidity function must be `payable`.
524      *
525      * _Available since v3.1._
526      */
527     function functionCallWithValue(
528         address target,
529         bytes memory data,
530         uint256 value
531     ) internal returns (bytes memory) {
532         return functionCallWithValue(target, data, value, "Address: low-level call with value failed");
533     }
534
535     /**
536      * @dev Same as {xref-Address-functionCallWithValue-address-bytes-uint256-}[`functionCallWithValue`], but
537      * with `errorMessage` as a fallback revert reason when `target` reverts.
538      *
539      * _Available since v3.1._
540      */
541     function functionCallWithValue(
542         address target,
543         bytes memory data,
544         uint256 value,
545         string memory errorMessage
546     ) internal returns (bytes memory) {
547         require(address(this).balance >= value, "Address: insufficient balance for call");
548         require(isContract(target), "Address: call to non-contract");
549
550         (bool success, bytes memory returndata) = target.call{value: value}(data);
```

QUICK REVIEW

Current Fees:
Buy: 13%
Sell: 19%

Owner can change Tx Limits.

Owner Privileges:

```
520        * Requirements:
521        *
522        * - the calling contract must have an ETH balance of at least `value`.
523        * - the called Solidity function must be `payable`.
524        *
525        * _Available since v3.1._
526        */
527       function functionCallWithValue(
528           address target,
529           bytes memory data,
530           uint256 value
531       ) internal returns (bytes memory) {
532           return functionCallWithValue(target, data, value, "Address: low-level call with value failed");
533       }
534
535       /**
536        * @dev Same as {xref-Address-functionCallWithValue-address-bytes-uint256-}[`functionCallWithValue`], but
537        * with `errorMessage` as a fallback revert reason when `target` reverts.
538        *
539        * _Available since v3.1._
540        */
541       function functionCallWithValue(
542           address target,
543           bytes memory data,
544           uint256 value,
545           string memory errorMessage
546       ) internal returns (bytes memory) {
547           require(address(this).balance >= value, "Address: insufficient balance for call");
548           require(isContract(target), "Address: call to non-contract");
549
550           (bool success, bytes memory returndata) = target.call{value: value}(data);
```

modifier onlyOwner() {
function renounceOwnership() public onlyOwner {
function transferOwnership(address newOwner) public onlyOwner {
function manualRebase() external onlyOwner{
function setAutomatedMarketMakerPair(address _pair, bool _value) public onlyOwner {
function setInitialDistributionFinished(bool _value) external onlyOwner {
function setFeeExempt(address _addr, bool _value) external onlyOwner {
function setTargetLiquidity(uint256 target, uint256 accuracy) external onlyOwner {
function setSwapBackSettings(bool _enabled, uint256 _num, uint256 _denom) external onlyOwner {
function setFeeReceivers(address _liquidityReceiver, address _treasuryReceiver, address _riskFreeValueReceiver)
external onlyOwner {
function clearStuckBalance(address _receiver) external onlyOwner {
function rescueToken(address tokenAddress, uint256 tokens) external onlyOwner returns (bool success){
function setAutoRebase(bool _autoRebase) external onlyOwner {
function setRebaseFrequency(uint256 _rebaseFrequency) external onlyOwner {
function setRewardYield(uint256 _rewardYield, uint256 _rewardYieldDenominator) external onlyOwner {
function setFeesOnNormalTransfers(bool _enabled) external onlyOwner {
function setIsLiquidityInBnb(bool _value) external onlyOwner {
function setNextRebase(uint256 _nextRebase) external onlyOwner {
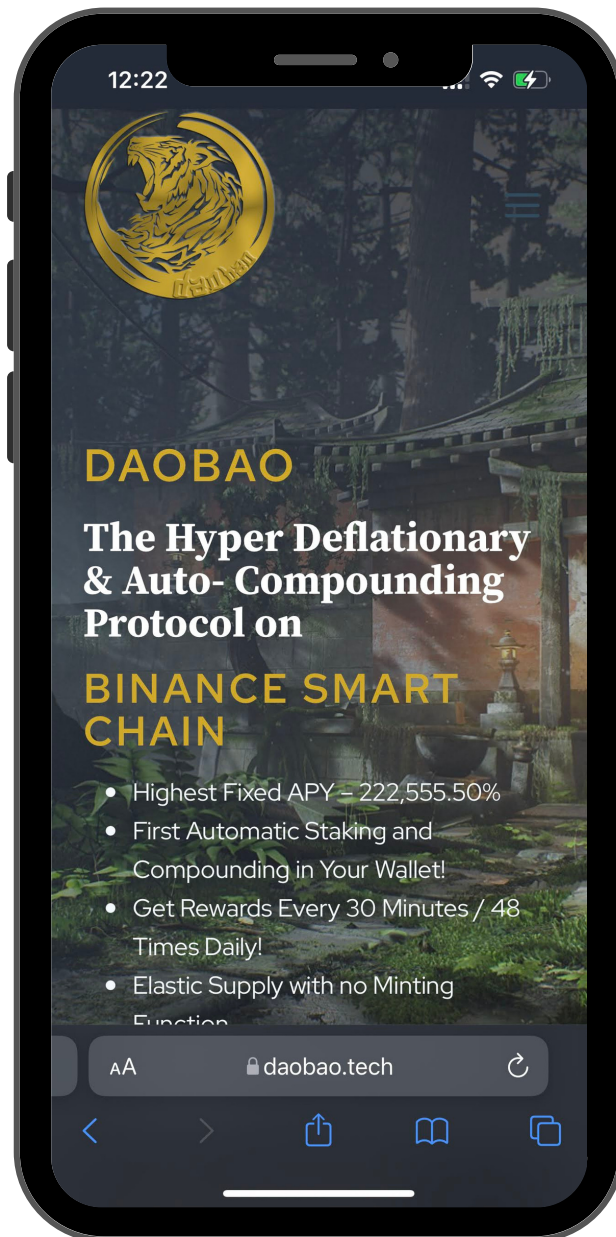function setMaxSellTransaction(uint256 _maxTxn) external onlyOwner {

Issue Checking:

N°	Issue description.	Checking Status.
1	Compiler Errors	Passed
2	Race conditions and Reentrancy. Cross-function race conditions.	Passed
3	Possible delays in data delivery.	Passed
4	Oracle calls.	Passed
5	Front running.	Passed
6	Timestamp dependence.	Passed
7	Integer Overflow and Underflow.	Passed
8	DoS with Revert.	Passed
9	DoS with block gas limit.	Passed
10	Methods execution permissions.	Passed
11	Economy model.	Passed
12	The impact of the exchange rate on the logic.	Passed
13	Private user data leaks.	Passed
14	Malicious Event log	Passed
15	Scoping and Declarations.	Passed
16	Uninitialized storage pointers.	Passed
17	Arithmetic accuracy.	Passed
18	Design Logic.	Passed
19	Cross-function race conditions.	Passed
20	Safe Zeppelin module	Passed
21	Fallback function security.	Passed

Audit Result:

PASSED

WEBSITE - OVERVIEW

⏱ Domain Registration: 2022-02-15

6 days old

SSL CERTIFICATE: A

(https://www.ssllabs.com)

SOCIAL MEDIA - OVERVIEW

DAOBAOFINANCE

DAOBAOFINANCE

DAOBAO.TECH